Build a binary text classifier using deep learning.

Dataset:
find dataset inside char-cnn/Deep-text-classification folder, it has a set of positive and negative examples. Deep-text-classification is in fact a full implementation of doing text classification using CNN.

Task:
You are going to use word2vec or glove pre-trained models to do the text classification.
Download the word2vec or glove dataset from below link.
Use this dataset (you don't need to retrain it) to build a text classifier. (load vec for each word from these pretrained words, combine vec in a certain way to produce a feature vector, use logistic regression to train a binary classifier)

Pretrained word2vec dataset link:
https://github.com/3Top/word2vec-api#where-to-get-a-pretrained-models

how to load the vectors:

```
filename = 'glove.6B.50d.txt'
def loadGloVe(filename):
vocab = []
embd = []
file = open(filename,'r')
for line in file.readlines():
row = line.strip().split(' ')
vocab.append(row[0])
embd.append(row[1:])
print('Loaded GloVe!')
file.close()
return vocab,embd
vocab,embd = loadGloVe(filename)
vocab_size = len(vocab)
embedding_dim = len(embd[0])
embedding = np.asarray(embd)
```

Useful references:
https://ireneli.eu/2016/06/14/tensorflow-06-word-embeddings-1/
https://ireneli.eu/2017/01/17/tensorflow-07-word-embeddings-2-loading-pre-trained-vectors/